WHAT IS CLAIMED IS:

1. For use in a wide-issue pipelined processor, a mechanism for reducing pipeline stalls between nested calls, comprising:

a program counter (PC) generator that generates return PC values for call instructions in a pipeline of said processor; and

return PC storage, coupled to said PC generator and located in an execution core of said processor, that stores said return PC values and makes ones of said return PC values available to a PC of said processor upon execution of corresponding return instructions.

2. The mechanism as recited in Claim 1 wherein said PC generator is associated with an instruction issue unit of said processor.

3. The mechanism as recited in Claim 1 wherein said PC generator generates each of said return PC values in a single clock cycle.

4. The mechanism as recited in Claim 1 wherein a return PC queue of said return PC storage has at least as many slots as a number of call instructions a fetch/decode stage of said pipeline can decode prior to grouping.

5. The mechanism as recited in Claim 1 wherein said return PC values move through registers of said return PC storage as corresponding ones of said return instructions move through stages in said pipeline.

6. The mechanism as recited in Claim 1 wherein said return PC storage makes said ones of said return PC values available to a PC of said processor as said corresponding return instructions are in an execution stage of said pipeline.

7. The mechanism as recited in Claim 1 wherein said call instruction is executed in a fetch/decode stage of said pipeline.

8. The mechanism as recited in Claim 1 wherein said processor is a digital signal processor.

9. For use in a wide-issue pipelined processor, a method of reducing pipeline stalls between nested calls, comprising:

generating return PC values for call instructions in a pipeline of said processor;

storing said return PC values in return PC storage located in an execution core of said processor; and

making ones of said return PC values available to a PC of said processor upon execution of corresponding return instructions.

10. The method as recited in Claim 9 wherein said generating is carried out in an instruction issue unit of said processor.

11. The method as recited in Claim 9 wherein said generating comprises generating each of said return PC values in a single clock cycle.

12. The method as recited in Claim 9 wherein a return PC queue of said return PC storage has at least as many slots as a number of call instructions a fetch/decode stage of said pipeline can decode prior to grouping.

13. The method as recited in Claim 9 further comprising moving said return PC values through registers of said return PC storage as corresponding ones of said return instructions move

through stages in said pipeline.

14. The method as recited in Claim 9 wherein said return PC storage makes said ones of said return PC values available to a PC of said processor as said corresponding return instructions are in an execution stage of said pipeline.

15. The method as recited in Claim 9 further comprising executing said call instruction in a fetch/decode stage of said pipeline.

16. The method as recited in Claim 9 wherein said processor is a digital signal processor.

17. A digital signal processor, comprising:

a pipeline having stages capable of executing call instructions;

a wide-issue instruction issue unit;

a program counter (PC) generator that generates return PC values for call instructions in a pipeline of said processor; and

return PC storage, coupled to said PC generator and located in an execution core of said processor, that stores said return PC values and makes ones of said return PC values available to a PC of said processor upon execution of corresponding return instructions.

18. The DSP as recited in Claim 17 wherein said PC generator is associated with an instruction issue unit of said DSP.

19. The DSP as recited in Claim 17 wherein said PC generator generates each of said return PC values in a single clock cycle.

20. The DSP as recited in Claim 17 wherein a return PC queue of said return PC storage has at least as many slots as a number of call instructions a fetch/decode stage of said pipeline can decode prior to grouping.

21. The DSP as recited in Claim 17 wherein said return PC values move through registers of said return PC storage as

corresponding ones of said return instructions move through stages in said pipeline.

22. The DSP as recited in Claim 17 wherein said return PC storage makes said ones of said return PC values available to a PC of said DSP as said corresponding return instructions are in an execution stage of said pipeline.

23. The DSP as recited in Claim 17 wherein said call instruction is executed in a fetch/decode stage of said pipeline.